Project screening: How to say “No” without hurting your career or your company

By Eduardo Miranda
ESCOM, May 2001, London

Abstract

Despite the large number of failed projects due to unrealistic schedules, their sponsors continue to push for development times that bear no relation to the productivity of their organizations, in the hope that next time things will be different. This paper postulates, that is possible to discriminate a challenging project from a death march one by comparing its footprint or profile to that of an idealized project of similar characteristics.

1. Introduction

Could you succeed at a project that would require you to:

- Bring together a team of 50 people in a month and a half?
- Integrate 100 interfaces in 40 days? and
- Complete more than 30 tasks, all of them on-time?

Although nobody benefits from failed projects, their sponsors continue to push for development times that bear no relation to the productivity of their organizations, in the hope that next time things will be different. In the meantime, developers, out of inexperience or fear, continue to agree and sometimes even foster unforgivable timetables. Reasonable schedules can only be attained if there is an equal understanding among all project stakeholders about the magnitude and complexity of the work to be done.

The three questions above focus on essential and highly visible attributes understandable to all project stakeholders: Can a project team be put together, can all the pieces be made to work in concert, and can a large number of tasks be consistently finished on-time? If a developing organization is not able or willing to fully staff a project, if the number of people is just too big for the team to jell on the time available, if there is no room to recover from a couple of late tasks, or if the are too many interfaces to integrate and keep under control, the project is not a *“challenge”,* it’s a *“Death March project”*[1].

The idea behind project screening, is to compare the characteristics of project you are about to take on, to the profiles of several idealized projects whose staff requirements, interface complexity and on-time probability can be calculated using few data and common sense expressions, and to use these profiles to decide whether you have a reasonable chance of success or not. The comparison between the requirements of an idealized and a real project is a valid one, since the distance that separates a *challenging* project from a *death march* project, is not 1 or 2 people or whether the probability of being on-time is 74% versus 75.5%, it is rather in the order of 25 to 40% of whatever a plausible estimate is.

The need for a method to supplement the expert judgement with some estimators of the true project complexity, has its root in the inability of the human mind to deal with non-linear phenomena[2], which as will see, result from just breaking down the system into one more subsystem to do more in parallel or from adding a couple of people to speed-up the development process.

The present article explains how these profiles can be constructed and gives an example of how they could be used.

2. The Project Mechanics

Process improvement is a win-win strategy, but usually it does not happen overnight. Neither does it yield the quantum-leap productivity increases many projects require to bridge the gap between what is wanted and what can be afforded. Therefore, other things being equal, the bigger the product or the shorter the time, the more people are needed.

Once the methods, tools and technical approach have been decided, there are only three basic trade-off a project manager can resort to:

- Time-Resources
- Time-Functionality
- Resources-Functionality

By extending the development time, the headcount requirement is reduced. Conversely, by adding more resources the development time is shortened. But as shown by Figure 1, the number of communication channels grows in geometric proportion to the team size, so as people are added to the project, the communication and coordination needs of the team increase, taking away part of the productivity contributed by each individual. This law of diminishing returns prevents the unconditional swap of months by people[3].

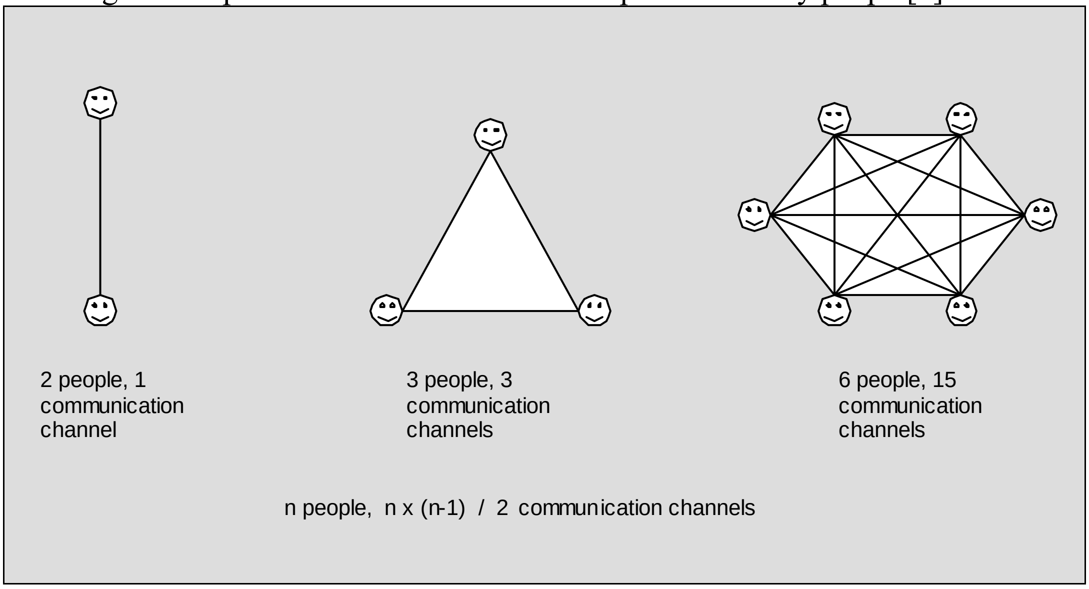

Figure 1 - Project communication

In reality, not everybody in a project communicates with everybody else. As illustrated by Figure 2, there are clusters of people where practically everyone communicates with everyone else and there are "gatekeepers" which communicates across clusters. It is not difficult to conclude, that these clusters correspond to people working closely together, for example into common subsystems. Therefore, a more realistic approximation of the number of channels in a project is given by a mixture of intra- and inter-team interactions as shown by Figure 3.

As we will see, the increase in communications overhead is not the only undesirable side effect of an increase in team size. It also affects the integration effort and the project's on-time probability.

3. Project Profiling

A project profile is an abstraction or idealization of the characteristics of a project. As the purpose here, is to quickly evaluate the feasibility and the risk of a project, I chose to profile a project in terms of the approximate number of people that will be involved, the number of interfaces it will need to handle and its probability of being completed on-time. Obviously, other profiles are possible, but one that includes these three variables is certainly appropriate for our stated purpose.

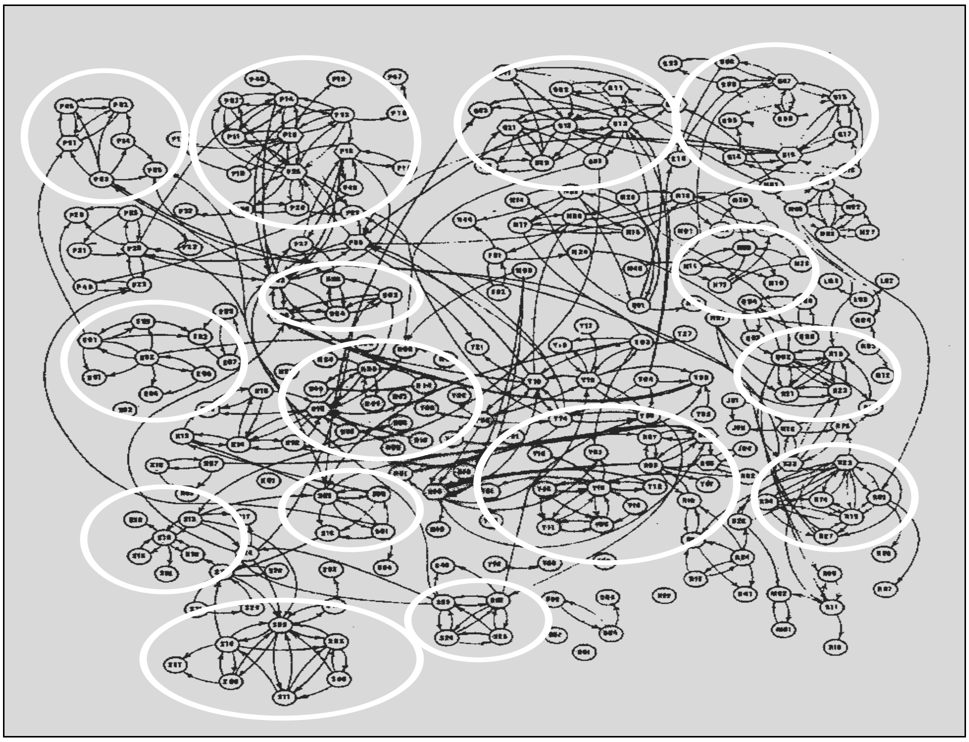

Figure 2 – Communications in an organization, T. Allen, MIT, Class Notes

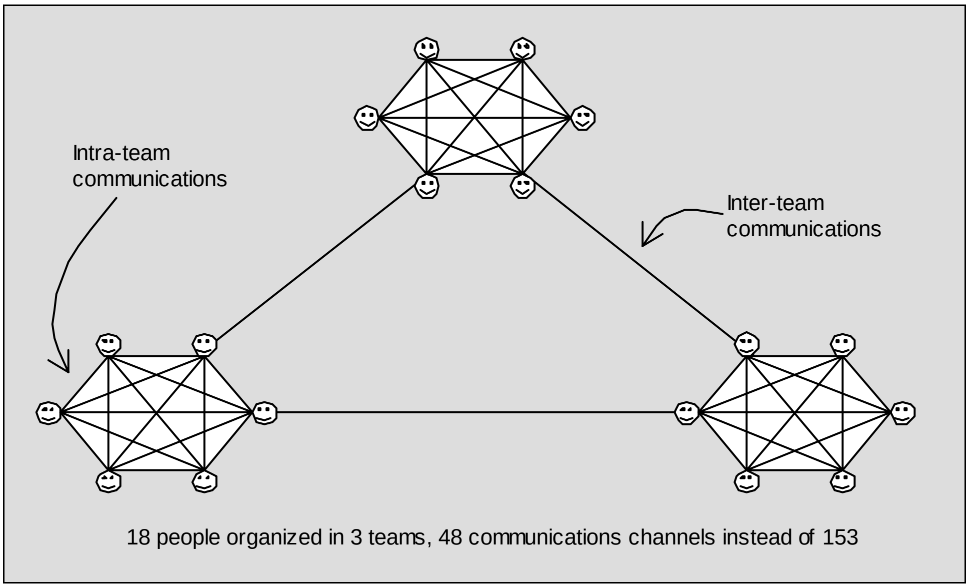

Figure 3 - A more realistic view of project communications

Paraphrasing J. Forrester[4], the impossibility of not knowing the exact values or relationships between all variables, should not preclude as from building useful models which could help us better understand complex problems and make better decisions. A project profile does not need to be exact to be useful, it will not be used for planning or tendering, it suffices that it answers the questions we post to it more precisely and accurately than alternatives methods and that is equally understandable to practitioners and laymen.

The following paragraphs define the calculations I chose to build the project profile. Three parameters: Uncertainty Assessment, Number of Subsystems and Communication Overhead are used to capture project-specific information.

The UncertaintyAssessment, express the variability that could be expected in the completion of a task, based on common risk factors. See Table 1 for a definition of common risks. A low value of this parameter means that the task is likely to be completed on time. The value should be jointly estimated by the project sponsor, the project manager and the system architect.

The NoSubsystems is a technical parameter, which reflects the breakdown of the system into cohesive parts as decided by the system architect.

The CommOverhead parameter captures the effort lost per interaction among members of the same or different teams. For example, if one engineer spends on average 30 minutes every time he communicates with one of his colleagues, the communication overhead parameter will be about 6%.

When looking at the specific calculations, please bear in mind that these expressions do not pretend to define, the probably un-calculable "true" value of the attribute, but rather to express meaningful relations between: project size, project schedule, staffing levels, integration complexity and on-time probabilities.

	Risk Factor		**Prob. of not completing the task on-time (%)**
	Initial uncertainty value[1]		5
If applicable	New product development	+	[2.5, 5]
	New technology / process being used	+	[2.5, 5]
	People capability / experience	+	[2.5, 5]
	Multiple organizations involved	+	[2.5, 5]
	Own risk factors	+	[2.5, 5]
	Uncertainty Assessment (%)	=	Σ risk factors

Table 1 - Uncertainty Assessment Table

3.1. Project Staff

Using a simple productivity number as a starting point, the approximate number of people needed by a project could be calculated using the equations shown in Figure 4.

3.2. Internal Interfaces

The number of Internal Interfaces is used as a surrogate for the integration and configuration management needed to put the system together. Its calculation is shown in Figure 5. As illustrated in Figure 6, for every combination of system size and development schedule, there is an optimal NoSubsystems that minimizes the number of internal interfaces.

$$ProjectStaff = NominalHeadcount + IntraTeamAdj + InterTeamAdj$$

$$NominalHeadcount = \frac{ApplicationSize}{Productivity * Schedule}$$

$$TeamSize = \frac{NominalHeadCount}{NoSubsystems}$$

$$IntraTeamAdj = \frac{TeamSize * (TeamSize - 1)}{2} * CommOverhead$$

$$InterTeamAdj = \frac{TeamSize * NoSubsystems * (NoSubsystems - 1)}{2} * CommOverhead$$

Figure 4 - Team size calculations

[1] These values could be used as guidelines, but they have no other meaning.

$$InternalInterfaces = NoSubsystems * ModuleInterfaces + SubsystemInterfaces$$

$$ModuleInterfaces = \frac{TeamSize * (TeamSize - 1)}{2}$$

$$SubsystemInterfaces = \frac{NoSubsystems * (NoSubsystems - 1)}{2}$$

Figure 5 – Interface calculations

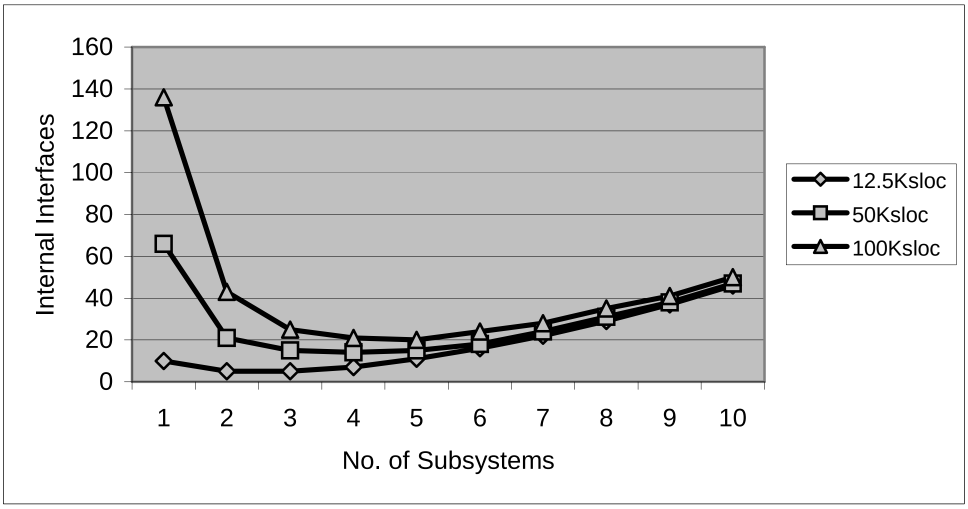

Figure 6 - Impact of system partitioning on the number of internal interfaces

3.3. On-time Probability

The On-time Probability, is a subjective probability reflecting the likelihood of completing the project in a time not exceeding, by more than 10% the scheduled development time, that is for a project scheduled to be 12 months long, to finish on or before 13.2 months. Figure 7 explain the meaning of the OnTimeProbability attribute. As with the other attributes, the OnTimeProbability should not be taken literally, but as an indication of the level of uncertainty to be expected in a project as the result of estimation errors, productivity assumptions, technology instability, etc.

The asymmetry of the triangular distribution chosen, reflects the existence of a minimum time beyond which it is almost impossible to complete the project, and the fact that there are very few things which could be done to shorten the schedule which are not discounted for at the outset of the project, while many that could go wrong are usually not accounted for[5]. The triangular distribution also captures the fact, that in a software development project most tasks' duration are not independent, but rather correlated through factors like complexity and team capability, so adding-up tasks' duration, does not result in a nice bell-shaped curve but rather on some skewed distribution that mimics Murphy's law!

Figure 8 depicts the work organization selected for the calculation of the project's OnTimeProbability, and Figure 9 the calculations themselves.

The standard deviation[2] of the tasks is assumed proportional to the following factors:

- Task's duration;
- UncertaintyAssessment;
- NoSubsystems in the case of the design activity;
- ProjectStaff in the case of the development; and
- InternalInterfaces in the case of the Integration

[2] σ Is the Greek letter used to denote the standard deviation of a random variable.

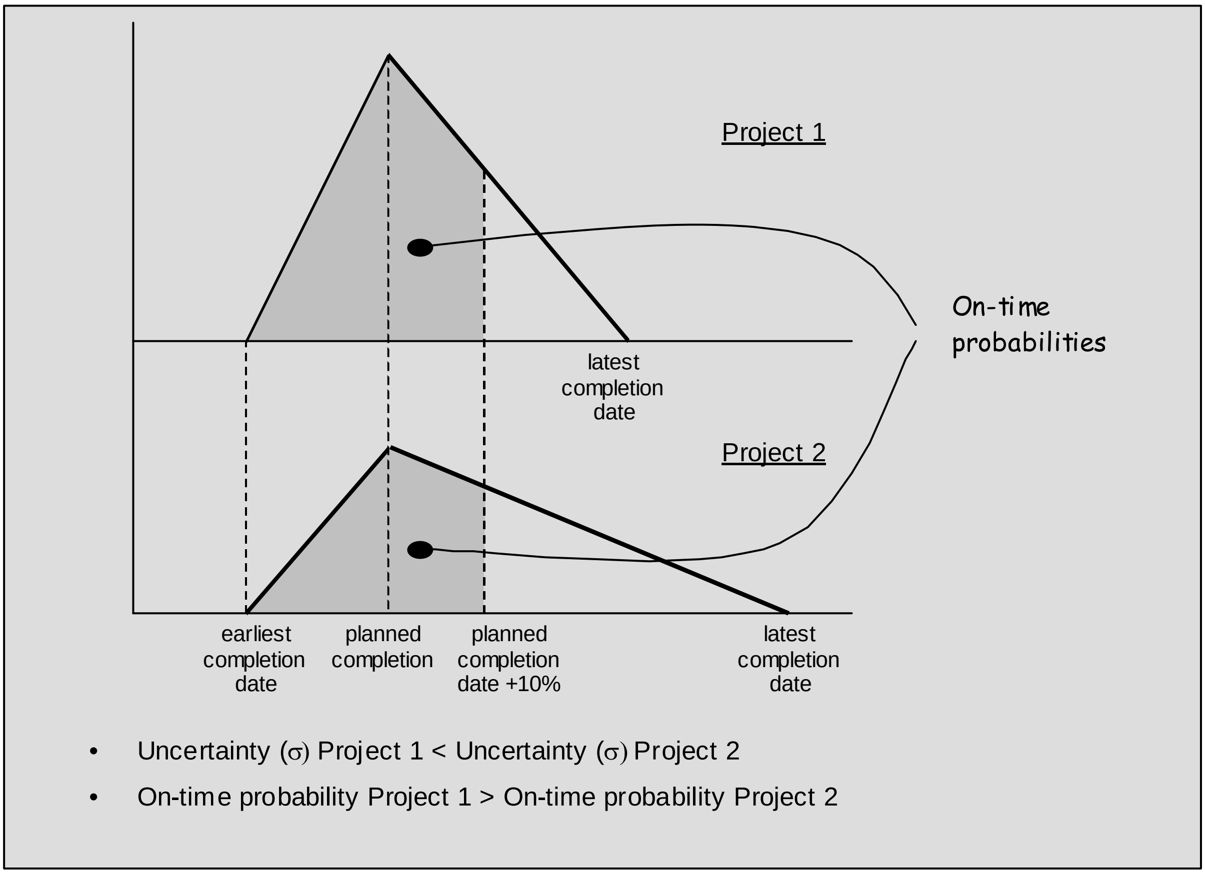

Figure 7 – The meaning of the OnTimeProbability

Each of the last three driving factors contributes 1%[3] of the task duration, times the magnitude of the factor, to its variability. For example, a 10 person project, lasting 12 months and having 3 subsystems will have an intrinsic variability of 2.7 days in design, 14.6 days in development and 19.1 in integration. Please notice that variability does not mean adding days to the tasks' duration, but rather that the duration of the task is a random variable whose actual value has a high probability, around 97%, of being comprised in the four σ interval existing between the earliest completion date and the latest completion date[4] of the task or project.

4. The Project Profiles Table

The Project Profiles Table (PPT), tabulates a number of project profiles for common combinations of project size, productivity and schedule, see Table 2. The PPT also includes references to previous developments carried out by the organization to be used as size and productivity clues. For example, Project "Y" size was 25 KSLOC[5] while Project "Z" was 75 KSLOC with productivity in the order of 500 and 250 KSLOC/MM respectively.

---

[3] This is a magic number. On its own it looks quite harmless, but when multiplied by the number of internal interfaces or the number of people in the project, it produces amazing results.
[4] Chebyshev's Theorem states that $P(|X-\mu| \geq k\sigma) \leq 1/k^2$. Although this might not be a tight bound in all cases, it is surprising that such a bound can be found to hold for all possible discrete and continuous distributions.
[5] The size units should reflect whatever is used by the organization, be it: SLOC, Function Points or Object Points. The key is understandability.

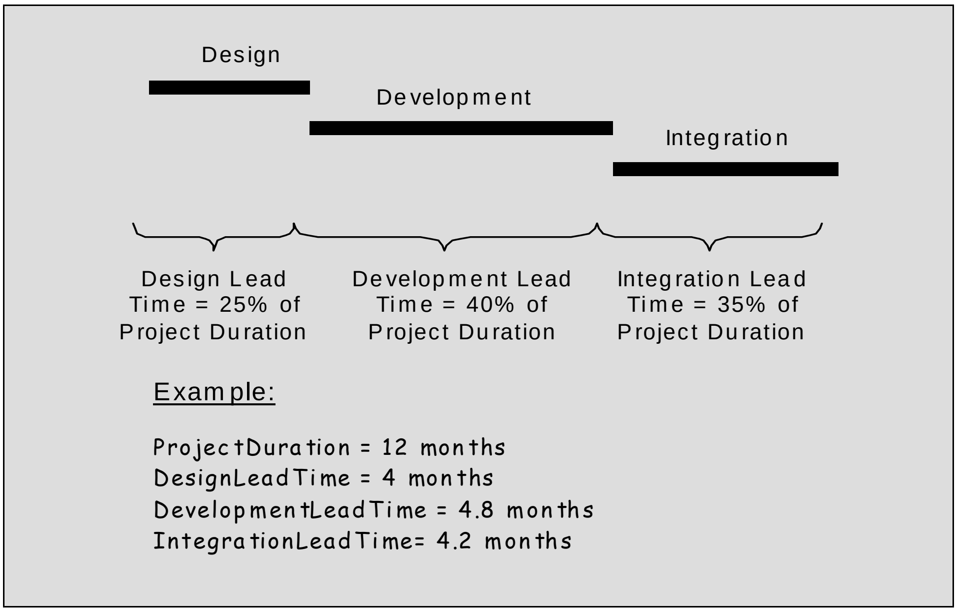

Figure 8 - Project model

$$OntimeProbability = 1 - \frac{(LatestCompletion - PlannedCompletion + 10\%)^2}{(LatestCompletion - EarliestCompletion) * (LatestCompletion - PlannedCompletion)}$$

$$EarliestCompletion = PlannedCompletion - \sigma_{Project}$$

$$LatestCompletion = PlannedCompletion + 3 * \sigma_{Project}$$

$$\sigma_{Project} = \sqrt{\sigma^2_{Design} + \sigma^2_{Development} + \sigma^2_{Integration}}$$

$$\sigma_{Design} = DesignLeadTime * (UncertaintyAssessment + NoSubsystems * .01)$$

$$\sigma_{Development} = DevelopmentLeadTime * (UncertaintyAssessment + ProjectStaff * .01)$$

$$\sigma_{Integration} = IntegrationLeadTime * (UncertaintyAssessment + InternalInterfaces * .01)$$

Figure 9 - Probability calculations

The PPT provides project stakeholders the means to screen-out death march projects and produce ballpark estimates by just looking at the attributes tabulated in it. The PPT is not an estimation tool, it is designed to be a negotiation tool and help you give an answer to your boss, when he or she request a “quick estimate” for this new hot project, that the Director of Finance or the marketing guys have just requested

5. Using the Project Profile Table

To illustrate the use of the PPT, a hypothetical dialogue between John, a Product Manager, and Caroline, a Project Manager in charge of developing a new product, will be used.

John – *“We need to have the new software ready by the end of the year”*

Caroline – *“That leaves us with only 6 months to complete the project”*

John – *“That’s our market window”*

Caroline – *“Judging from the product spec, the new software is going to be bigger than Application Y and smaller than Z, let’s say around 50 KSLOC, so if we want to have it in six months, with the resources available, we will have to perform at least as good as we did in Project Y.”*

John – *“What about putting more people into it? We need to get the new release ready for December 23rd.”*

Caroline (looking at Table 1) – *"We' ll need around 22 people. We could steal some of them from another project, but we still have to test and integrate roughly 47 internal interfaces, and our overall probability of success will be about 43%, less than 50/50 chance. Do you want to bet our bonus and, probably our careers on that?"*

John – " *What is the basis of your estimate?"*

Caroline – *"I am assuming we break the system down into 5 or 6 subsystems that we can develop in parallel, and that there is not a lot of unknowns, let's say around 15% due to problems with the new midleware and some changes in marketing."*

John – *"So, what do you propose?"*

Caroline – *"Well, we either perform twice as better as Project Y by putting our best people into it, I think we could do somehow better, but I doubt we can double it, or we can cut the software size down to the 25-30 KSLOC range. In both cases we' ll need around 11 people and our overall probability of success will be about 53%"*

John – *"It doesn't look good, does it? …."*

Caroline – *"If we could trim some gold-plated features, and put our best people, I would feel much more confident."*

John – *"OK, will do. But we better get started, you go do the detailed planning, while I go to talk to the marketing guys, we are already one hour late!"*

	Schedule & Productivity	Size **25**	Project "Y"		Size **50**			Size **75**	Project "Z"	
	6 months	Proj. Staff	Internal Interfaces	On time probability	Proj. Staff	Internal Interfaces	On time probability	Proj. Staff	Internal Interfaces	On time probability
	250	**22**	47	43%	**50**	235	30%	**83**	657	27%
Proj. Y	500	**11**	17	53%	**22**	47	43%	**35**	115	35%
	1000	**6**	11	58%	**11**	17	53%	**16**	28	48%
	2000	**3**	11	59%	**6**	11	58%	**8**	12	56%
	12 months									
	250	**11**	17	53%	**22**	47	43%	**35**	115	35%
	500	**6**	11	58%	**11**	17	53%	**16**	28	48%
	1000	**3**	11	59%	**6**	11	58%	**8**	12	56%
	2000	**2**	11	60%	**3**	11	59%	**4**	11	59%
	18 months									
Proj. Z	250	**7**	11	57%	**15**	25	49%	**22**	47	43%
	500	**4**	11	59%	**7**	11	57%	**11**	17	53%
	1000	**2**	11	60%	**4**	11	59%	**6**	11	58%
	2000	**1**	11	60%	**2**	11	60%	**3**	11	59%

Table 2 - Project Profile Table, Uncertainty Assessment = 15%, No. of Subsystems = 5, Communication Overhead =10%, the highlighted values are used on the example bellow.

The use of the PPT could be complemented with that of two other tables (Table 1 and 3), which add objectivity to the process:

- The "Uncertainty Assessment Table" and
- The "Go-ahead Decision Table".

These two tables provide simple heuristics about how to quantify risk factors and action guidelines for those deciding the birth of a project.

The numbers on the Uncertainty Assessment Table express the probability of not completing individual tasks on-time due to common risk factors. The use of very small probabilities makes it difficult for anyone to disagree with them in isolation. The surprise usually comes when they are combined and compounded throughout a large number of tasks.

The Go-ahead Decision Table is read from left to right along the gray scales until a conclusion is reached, i.e. “Go-ahead” or “Renegotiate the project”. The conditions contained in each column under the headings “People”, “Internal Interfaces” and “On-time Probability” should be asked and answered using the values provided by the PPT for the project being screened. For example, if the PPT specifies a staff of 35 people, the questions are: Do I have 35 people available to work on this project? Can I make them work as a team in six months?, etc.

<table>
<tr><th>People</th><th>Internal Interfaces</th><th>On-time Probability</th><th>Criticality of delivery date</th><th>Action</th></tr>
<tr><td rowspan="5">Have resources
Able to build team
Able to manage team
Enough leaders to co-ordinate</td><td rowspan="4">Able to manage
Able to test
Able to integrate</td><td>> 50%</td><td></td><td>Go-ahead</td></tr>
<tr><td rowspan="2">35 < p <=50%</td><td>High</td><td>Renegotiate Schedule
Reduce Functionality
Establish contingency plans
Improve productivity</td></tr>
<tr><td>Low</td><td>Go-ahead</td></tr>
<tr><td>p <=35 %</td><td></td><td>Renegotiate Schedule
Reduce Functionality
Improve productivity</td></tr>
<tr><td>No to any of the above</td><td></td><td></td><td>Renegotiate Schedule
Reduce Functionality
Improve productivity</td></tr>
<tr><td>No to any of the above</td><td></td><td></td><td></td><td>Renegotiate Schedule
Reduce Functionality
Improve productivity</td></tr>
</table>

Table 3 - Go-ahead guidelines

6. Summary

The John and Caroline story has a happy ending, but it’s not always like that. Too many careers, personal lives, and businesses have been hurt because of unrealistic expectations. By promoting a mutual understanding between project stakeholders about the complexity and magnitude of the job to be done, the PPT helps project managers and sponsors avoid committing to impossible schedules.

In putting together a PPT for your organization, privilege simplicity and visibility over unnecessary and unattainable accuracy. Remember that the miscalculations at the root of a Death March project are those off by more than 25 percent.

Taking into consideration all the simplifications made in calculating team sizes, number of internal interfaces and on-time probabilities, the PPT could be regarded as a very favorable scenario. Therefore, if you do not feel comfortable with the numbers shown in it, reality will not make the job any easier. Good luck with your next project.

7. Acknowledgements

Thanks to Sylvain Lemire, Sylvain Loranger, Gaetano Lombardi and Tamara Keating for their comments and insight.

8. References

[1] Death March, The Complete Software Developer’s Guide to Surviving “Mission Impossible” Projects, E. Yourdon, Prentice-Hall, 1997

[2] The Logic of Failure, D. Doner, Addison-Wesley, 1996

[3] The Mythical Man-Month, F. Brooks, Addison–Wesley, 1975

[4] J. W. Forrester, Industrial Dynamics, The MIT Press, 1961

[5] Practical Risk Assessment for Project Management, S. Grey, John Willey & Sons, 1995